

WHAT IS CLAIMED IS:

1. A data storage system comprising:

N servers, where $N \geq 2$ and is an integer;

D disks, where $D \geq 2$ and is an integer, each server in communication with each disk, each disk having a reserved disk block for each of the N servers; and

a disk arbitration mechanism that uses a timestamp-based voting algorithm over the disk blocks associated with the servers to exchange votes for a primary server to arbitrate access of the servers to a set of disks.

2. A data storage system comprising:

N servers, where $N \geq 2$ and is an integer;

D disks, where $D \geq 2$ and is an integer, each server in communication with each disk, each disk having a reserved disk block for each of the N servers; and

a disk arbitration mechanism where each of the N servers writes its state in its own associated disk block in each disk, and reads all the other servers' disk blocks in each disk in order to determine which server has access to, and use and control of the disks at a given time.

3. A system as described in Claim 2 wherein each server has an index.

0953336-03001

4. A system as described in Claim 3 wherein the disk arbitration mechanism causes each server at first predetermined times to read all of the disk blocks, and write its own disk block to determine which server has access to, and use and control of the disks at a given time.

5. A system as described in Claim 4 wherein each server includes a state machine and a local RAM, and maintains in local RAM a last time at which each servers' state changed and a value associated with the state when it last changed.

6. A system as described in Claim 5 wherein each server determines which of the other servers are operating by identifying which of the other servers had their state change during second predetermined times.

7. A method for storing data comprising the steps of:

writing by N of servers into each servers' own associated disk block in each disk of D disks its state, where $N \geq 2$ and $D \geq 2$ and are integers; and

reading by each server all the other servers' disk blocks in each disk in order to determine which server has access to, and use and control of the disks at a given time.

8. A method as described in Claim 7 wherein the reading step includes the step of performing a voting protocol to determine which server has access to, and use and control of the disks at a given time.

9. A method as described in Claim 8 including after the reading step, there are the steps of determining which server

becomes a winning server and has access to, and use and control of the disk at a given time; and accessing the disk exclusively by the winning server.

10. A method as described in Claim 9 wherein the accessing step includes the step of transmitting by the winning server its state from not winning to winning and invalidating by the winning server all caches of the winning server.

11. A method as described in Claim 10 wherein the writing step includes the step of assigning an index to each server.

12. A method as described in Claim 11 wherein the reading step includes the step of reading at predetermined times by each server all disk blocks and writing its own respective disk block.

13. A method as described in Claim 12 wherein the writing step includes the step of maintaining by each server in each servers' own local RAM a last time for each other server when each other servers' status changed and a value of a status counter at the last time.

14. A method as described in Claim 13 wherein the reading step includes the step of determining by each server which of the other servers are operating by declaring that each of the other servers whose status has changed within a last predetermined time period is operating.

15. A method as described in Claim 14 wherein the reading step includes the step of voting by the servers that are up

16. A method as described in Claim 15 wherein the reading step includes the step of voting for the server that is up and has a lowest index if no server believes it is the winning server.